

## **Nursing Facility MDS Record Selection and Case Mix Calculation SFY 2006, Quarter 3 (January 2006)**

The following is a brief overview of the steps involved in calculating the case mix for each NF beginning in State FY2006 Q2:

1. Obtain an updated list of Medicaid NF residents with an add-on rate from the Resident Assessment team. Import new records into the “AddOnRateRecipients” table.
  2. Make a copy of the previous quarter’s case mix calculation database for use in the new period – rename it to fit the new period.
  3. Clear prior period data tables in the new (copied) DB.
    - a. MdsAllData
    - b. MdsAllData\_Excluded\_IntensiveSkilled
    - c. MdsAllData\_ExcludedAssessments\_BadData
  4. Append MDS data into the “MdsAllData” table and Update.
    - a. Append Discharge records to the “MdsAllData” table.
    - b. Append Assessment records to the “MdsAllData” table.
    - c. Update corrected assessments to Null values – except Assessment Internal ID.
      - i. This will remove assessments that have subsequently been corrected from the table so they are not counted twice.
    - d. Update Intensive Skilled assessment records to Null values – except Assessment Internal ID.
      - i. Remove records from the “MdsAllData” table if the assessment occurs while the person is receiving an “ISC” add-on rate amount. Append the removed records to the “MdsAllData\_Excluded\_IntensiveSkilled” table for reference.
  5. Calculate the RUG days by entering the “period of interest” dates on the MS Access form and clicking the calculate days button.
    - a. Calculate Days button – This button will calculate the individual RUG days by executing Visual Basic code (see Appendix A) for the period of interest against the “qryGetData” query which contains the following data:
      - i. FacilityMedicaidID (Sort1)
        1. Filter:
          - a. Is Not Null And <>””

- ii. SSN (Sort2)
    - 1. This is a key sort field, so it is essential that it be filled in correctly.
    - 2. Filter:
      - a. Is Numeric
      - b. Is 9-digits long
  - iii. AdmitDate
    - 1. For this field use the later of AB1 or A4A. (Previously, only AB1 was used.)
    - 2. This is a calculated field and is determined as follows:
      - a. AdmitDate: IIf(Not (IsNull(MdsAllData!AdmtDate\_AB1)) And Not (IsNull(MdsAllData!A4A\_Reentry)), nz(IIf(MdsAllData!AdmtDate\_AB1>MdsAllData!A4A\_Reentry, MdsAllData!AdmtDate\_AB1,MdsAllData!A4A\_Reentry)), IIf(Not (IsNull(MdsAllData!AdmtDate\_AB1)), MdsAllData!AdmtDate\_AB1, MdsAllData!A4A\_Reentry))
  - iv. AssessmentDate (R2B or R4) (Sort3)
    - 1. This is a calculated field and is determined as follows:
      - a. AssessmentDate: IIf(IsNull(MdsAllData.R2B\_AssessCompDT), MdsAllData.DischargeDate, MdsAllData.R2B\_AssessCompDT)
  - v. DischargeDate (R4)
  - vi. StateCalculatedMedicaidRUG
  - vii. AssessmentInternalID (Sort4)
  - viii. MedicareCodeQualifier (AA8B)
  - ix. ResidentMedicaidID (AA7)
  - x. Caid – True/False
    - 1. This is a calculated field and is determined as follows:
      - a. IIf((MdsAllData!MedicareCodeQualifier="6" Or MdsAllData!MedicareCodeQualifier Like " "), (IIf((Left(MdsAllData!ResidentMedicaidID,1)+"+" Or Len(MdsAllData!ResidentMedicaidID)=10), True, False)), False)
  - b. The results of this VB script are written to the “tblResults” table.
6. Validate that no resident has more total days than is possible in the period of interest. If a resident is found to have more days, then remove that residents data from the “MdsAllData” table and append it to the “MdsAllData\_ExcludedAssessments\_BadData” table and rerun the case mix calculation as noted in #4 above.
7. Sum the RUG days by RUG as follows:
- a. FacilityName (Group By)

b. FacilityMedicaidID	(Group By)
c. RUG	(Group By)
d. Weight	(Group By)
e. Days	(Sum)
f. WgtXDays: [Weight]*[Days]	(Expression)

8. Sum the RUG days by facility as follows:

a. FacilityMedicaidID	(Group By)
b. FacilityName	(Group By)
c. TotalDays: Sum(tblResults_DaysRugs_Sum!Days)	(Expression)
d. SumWgtXDays: WgtXDays	(Sum)
e. CaseMix: [SumWgtXDays]/[TotalDays]	(Expression)

#### Notes:

- This version of the Case Mix calculation uses non-Medicaid assessment data only for a prior assessment/discharge date(s) and/or the prior assessment RUG score.
- MDS Section AA, question 8b (Codes for assessments required for Medicare PPS or the State) – Any value other than “6” or Null (blank or a space) is a Medicare assessment and would not be included in a Medicaid case mix calculation.
- ResidentMedicaidID – In addition to AA8B, this field is used to help in determining if a resident is Medicaid eligible.

Minimum Data Set (MDS) – Version 2.0 instructions for Section AA, question 7 states the following: “Medicaid No. [“+” if pending, “N” if not a Medicaid recipient].” There is a further note, via asterisk, that the field is a key item for computerized resident tracking.

We have noticed quite a variety of data in this field that do not follow the instructions for data entry. (Examples: APPLY, D, N/A, NONE, NON3E, P, PENDING, ZHC...) We have also noted many instances where the SSN was used in the ResidentMedicaidID field. **To be considered a Medicaid assessment record, the following must be true:**

- AA8B is “6” or Null **and**
- ResidentMedicaidID is a 10-digit numeric or a “+” (the quotation marks would not actually be included in the field).
- Also refer to 5.a.x.1.a above for the syntax in determining whether an assessment is considered a Medicaid record.

## **RUG Calculation Translation:**

Is it a new person?

No –

Is there an admit date?

Yes –

Is the admit date later than the last assessment date?

Yes –

Is there a current RUG?

Yes – Use the current RUG

No – Use a default value

No –

Is there a previous RUG?

Yes – Use the previous RUG

No – Use a default value

No –

Is there a previous RUG?

Yes – Use the previous RUG

No – Use a default value

Yes –

Is there a current RUG?

Yes – Use the current RUG

No – Use a default value

## Appendix A – MDS Case Mix Calculation Syntax

- Use Medicaid records and prior – non-Medicaid payor – records as appropriate for RUG
  - Use the maximum date between Admit and the last Assessment/Discharge date
- 

### This routine writes the data to the results table.

```
Private Sub WriteResult(ByVal strFacility_toWrite As String, ByVal strSSN_toWrite As String, ByVal dtmBegin As Date, ByVal dtmEnd As Date, ByVal intDays As Integer, ByVal strRUG_toWrite As String, ByVal strAssessmentID_toWrite As String)
    DoCmd.RunSQL "INSERT INTO tblResults(FacilityMedicaidID, SSN, dtmBegin, dtmEnd, intDays, strRUG, strAssessmentId)VALUES('" & strFacility_toWrite & "', '" & strSSN_toWrite & "', #" & dtmBegin & "#, #" & dtmEnd & "#, " & intDays & ", '" & strRUG_toWrite & "', '" & strAssessmentID_toWrite & "')"
End Sub
```

---

### This function determines the appropriate RUG to use.

```
Private Function DetermineRUG(ByVal blnNew As Boolean, ByVal dtmAdmit As Date, ByVal dtmAssessmentLast As Date, ByVal strRUG As String, ByVal strRUGLast As String, ByVal strDefault As String) As String
    If Not blnNew Then
        If Not IsNull(dtmAdmit) Then
            If dtmAdmit > dtmAssessmentLast Then
                If strRUG <> "" Then
                    DetermineRUG = strRUG
                Else
                    DetermineRUG = strDefault
                End If
            Else
                If strRUGLast <> "" Then
                    DetermineRUG = strRUGLast
                Else
                    DetermineRUG = strDefault
                End If
            End If
        Else
            If strRUGLast <> "" Then
                DetermineRUG = strRUGLast
            Else
                DetermineRUG = strDefault
            End If
        End If
    Else
        If strRUGLast <> "" Then
            DetermineRUG = strRUGLast
        Else
            DetermineRUG = strDefault
        End If
    End If
End Function
```

```

        End If
    End If
Else
    If strRUG <> "" Then
        DetermineRUG = strRUG
    Else
        DetermineRUG = strDefault
    End If
End If
End Function

```

---

**This routine calculates the case mix days, etc.**

```

Private Sub Command8_Click()
Dim rs, rs2 As ADODB.Recordset
Dim strFacility, strFacilityLast, strSSN, strSSNLast, strRUG, strRUGLast, strRUGFinal, strAssessmentId, strAssessmentIdLast As String
Dim dtmAssessment, dtmAssessmentLast, dtmAdmit, dtmBegin, dtmEnd, dtmDischarge, dtmDischargeLast As Date
Dim intDays As Integer
Dim blnCaid, blnCaidLast, blnClose, blnWrite, blnNew, blnPause As Boolean
Set rs = New ADODB.Recordset
Set rs2 = New ADODB.Recordset
rs.Open "qryGetData", CurrentProject.AccessConnection, adOpenForwardOnly, adLockReadOnly
DoCmd.SetWarnings (WarningsOff)
DoCmd.RunSQL "DELETE FROM tblResults"
DoCmd.RunSQL "DELETE FROM tblAnomalies"
strRUGLast = ""
While Not rs.EOF
    strFacility = rs.Fields("FacilityMedicaidID")
    strSSN = rs.Fields("SSN")
    dtmAdmit = rs.Fields("AdmitDate")
    If IsNull(dtmAdmit) Then
        dtmAdmit = #12:00:00 AM#
    End If
    dtmAssessment = rs.Fields("AssessmentDate")
    dtmDischarge = rs.Fields("DischargeDate")
    strRUG = rs.Fields("StateCalculatedMedicaidRUG")
    If IsNull(strRUG) Then
        strRUG = ""
    End If
    strAssessmentId = rs.Fields("AssessmentInternalID")
    blnCaid = CBool(rs.Fields("Caid"))

```

```

If IsNumeric(strSSN) And Len(strSSN) = 9 Then
    'Is this a new person-facility pair?
    If strFacility = strFacilityLast And strSSN = strSSNLast Then
        blnNew = False
    Else
        blnNew = True
        'Eliminate bad data
        'If blnAnomaly Then
            ' DoCmd.RunSQL "DELETE FROM tblResults WHERE FacilityMedicaidID = "" & strFacilityLast & "" AND SSN = "" & strSSNLast & """
            ' DoCmd.RunSQL "INSERT INTO tblAnomalies SELECT MdsAllData.FacilityMedicaidID, MdsAllData.SSN, MdsAllData.AdmitDate,
IIf(IsNull(MdsAllData.AssessmentDate),MdsAllData.DischargeDate,MdsAllData.AssessmentDate) AS AssessmentDate, MdsAllData.DischargeDate,
MdsAllData.StateCalculatedMedicaidRUG, MdsAllData.AssessmentInternalID, MdsAllData.MedicareCodeQualifier, MdsAllData.ResidentMedicaidID,
IIf((MdsAllData!MedicareCodeQualifier='6' Or MdsAllData!MedicareCodeQualifier Like ' ),(IIf((MdsAllData!ResidentMedicaidID Like '+%' Or
Len(MdsAllData!ResidentMedicaidID)=10),True,False)),False) AS Caid FROM MdsAllData WHERE (((MdsAllData.FacilityMedicaidID) = "" & strFacilityLast & "") AND
((MdsAllData.SSN) = "" & strSSNLast & ""))
            ' blnAnomaly = False
        'End If
    End If

'did the assessment occur in the middle of the period? If so, close out the period
If blnNew And blnClose And dtmDischargeLast = #12:00:00 AM# Then
    dtmBegin = DateAdd("d", 1, dtmAssessmentLast)
    dtmEnd = Me.tbPeriodEnd.Value
    intDays = DateDiff("D", dtmBegin, dtmEnd) + 1
    strFacility_toWrite = strFacilityLast
    strSSN_toWrite = strSSNLast
    strRUG_toWrite = strRUGLast
    strAssessmentID_toWrite = strAssessmentID_Last
    Call WriteResult(strFacility_toWrite, strSSN_toWrite, dtmBegin, dtmEnd, intDays, strRUG_toWrite, strAssessmentID_toWrite)
End If
blnClose = False

'Is this a Medicaid Record?

If blnCaid Then
    blnWrite = False
    'Is this the same person-facility pair?
    If Not blnNew Then
        'Is there an Admit Date?
        If dtmAdmit <> #12:00:00 AM# Then

```

```

If dtmAdmit < dtmAssessmentLast Then
    dtmAdmit = dtmAssessmentLast
End If
'Is the Admit Date in our range of interest?
If dtmAdmit >= Me.tbPeriodBegin.Value And dtmAdmit <= Me.tbPeriodEnd.Value Then
    'Is the Assessment Date in our range of interest?
    If dtmAssessment >= Me.tbPeriodBegin.Value And dtmAssessment <= Me.tbPeriodEnd.Value Then
        'First(N)Admit(Y)AdmitInRange(Y)AssessInRange(Y)
        dtmBegin = dtmAdmit
        dtmEnd = dtmAssessment
        blnClose = True
        intDays = DateDiff("D", dtmBegin, dtmEnd)
        strFacility_toWrite = strFacility
        strSSN_toWrite = strSSN
        strRUG_toWrite = DetermineRUG(blnNew, dtmAdmit, dtmAssessmentLast, strRUG, strRUGLast, "NYYY")
        blnWrite = True
        'End of Line
    ElseIf dtmAssessment > Me.tbPeriodEnd.Value Then
        'First(N)Admit(Y)AdmitInRange(Y)AssessInRange(N)
        dtmBegin = dtmAdmit
        dtmEnd = Me.tbPeriodEnd.Value
        intDays = DateDiff("D", dtmBegin, dtmEnd)
        strFacility_toWrite = strFacility
        strSSN_toWrite = strSSN
        strRUG_toWrite = DetermineRUG(blnNew, dtmAdmit, dtmAssessmentLast, strRUG, strRUGLast, "NYYN")
        blnWrite = True
        'End of Line
    End If
ElseIf dtmAdmit < Me.tbPeriodBegin.Value Then
    'Is the Assessment Date in our range of interest?
    If dtmAssessment >= Me.tbPeriodBegin.Value And dtmAssessment <= Me.tbPeriodEnd.Value Then
        'First(N)Admit(Y)AdmitInRange(N)AssessInRange(Y)
        dtmBegin = Me.tbPeriodBegin.Value
        dtmEnd = dtmAssessment
        blnClose = True
        intDays = DateDiff("D", dtmBegin, dtmEnd)
        strFacility_toWrite = strFacility
        strSSN_toWrite = strSSN
        strRUG_toWrite = DetermineRUG(blnNew, dtmAdmit, dtmAssessmentLast, strRUG, strRUGLast, "NYNY")
        blnWrite = True
        'End of Line
    ElseIf dtmAssessment > Me.tbPeriodEnd.Value Then

```

```

'First(N)Admit(Y)AdmitInRange(Y)AssessInRange(N)
dtmBegin = Me.tbPeriodBegin.Value
dtmEnd = Me.tbPeriodEnd.Value
intDays = DateDiff("D", dtmBegin, dtmEnd)
strFacility_toWrite = strFacility
strSSN_toWrite = strSSN
strRUG_toWrite = DetermineRUG(blnNew, dtmAdmit, dtmAssessmentLast, strRUG, strRUGLast, "NYYN")
blnWrite = True
'End of Line
End If
End If
Else
    'Assign the Admit Date to the Last Assessment Date
    dtmAdmit = dtmAssessmentLast
    'Is the Admit Date in our range of interest?
    If dtmAdmit >= Me.tbPeriodBegin.Value And dtmAdmit <= Me.tbPeriodEnd.Value Then
        'Is the Assessment Date in our range of interest?
        If dtmAssessment >= Me.tbPeriodBegin.Value And dtmAssessment <= Me.tbPeriodEnd.Value Then
            'First(N)Admit(N)AdmitInRange(Y)AssessInRange(Y)
            dtmBegin = dtmAdmit
            dtmEnd = dtmAssessment
            blnClose = True
            intDays = DateDiff("D", dtmBegin, dtmEnd)
            strFacility_toWrite = strFacility
            strSSN_toWrite = strSSN
            strRUG_toWrite = DetermineRUG(blnNew, dtmAdmit, dtmAssessmentLast, strRUG, strRUGLast, "NNYY")
            blnWrite = True
            'End of Line
        ElseIf dtmAssessment > Me.tbPeriodEnd.Value Then
            'First(N)Admit(N)AdmitInRange(Y)AssessInRange(N)
            dtmBegin = dtmAdmit
            dtmEnd = Me.tbPeriodEnd.Value
            intDays = DateDiff("D", dtmBegin, dtmEnd)
            strFacility_toWrite = strFacility
            strSSN_toWrite = strSSN
            strRUG_toWrite = DetermineRUG(blnNew, dtmAdmit, dtmAssessmentLast, strRUG, strRUGLast, "NYNY")
            blnWrite = True
            'End of Line
        End If
    ElseIf dtmAdmit < Me.tbPeriodBegin.Value Then
        'Is the Assessment Date in our range of interest?
        If dtmAssessment >= Me.tbPeriodBegin.Value And dtmAssessment <= Me.tbPeriodEnd.Value Then

```

```

'First(N)Admit(N)AdmitInRange(N)AssessInRange(Y)
dtmBegin = Me.tbPeriodBegin.Value
dtmEnd = dtmAssessment
blnClose = True
intDays = DateDiff("D", dtmBegin, dtmEnd)
strFacility_toWrite = strFacility
strSSN_toWrite = strSSN
strRUG_toWrite = DetermineRUG(blnNew, dtmAdmit, dtmAssessmentLast, strRUG, strRUGLast, "NNNY")
blnWrite = True
'End of Line
ElseIf dtmAssessment > Me.tbPeriodEnd.Value Then
'First(N)Admit(N)AdmitInRange(N)AssessInRange(N)
dtmBegin = Me.tbPeriodBegin.Value
dtmEnd = Me.tbPeriodEnd.Value
intDays = DateDiff("D", dtmBegin, dtmEnd)
strFacility_toWrite = strFacility
strSSN_toWrite = strSSN
strRUG_toWrite = DetermineRUG(blnNew, dtmAdmit, dtmAssessmentLast, strRUG, strRUGLast, "NNNN")
blnWrite = True
'End of Line
End If
End If
Else
Is there an Admit Date?
If dtmAdmit <> #12:00:00 AM# Then
    Is the Admit Date in our range of interest?
    If dtmAdmit >= Me.tbPeriodBegin.Value And dtmAdmit <= Me.tbPeriodEnd.Value Then
        Is the Assessment Date in our range of interest?
        If dtmAssessment >= Me.tbPeriodBegin.Value And dtmAssessment <= Me.tbPeriodEnd.Value Then
            'First(Y)Admit(Y)AdmitInRange(Y)AssessInRange(Y)
            dtmBegin = dtmAdmit
            dtmEnd = dtmAssessment
            blnClose = True
            intDays = DateDiff("D", dtmBegin, dtmEnd)
            strFacility_toWrite = strFacility
            strSSN_toWrite = strSSN
            strRUG_toWrite = DetermineRUG(blnNew, dtmAdmit, dtmAssessmentLast, strRUG, strRUGLast, "YYYY")
            blnWrite = True
            'End of Line
        ElseIf dtmAssessment > Me.tbPeriodEnd.Value Then
            'First(Y)Admit(Y)AdmitInRange(Y)AssessInRange(N)

```

```

dtmBegin = dtmAdmit
dtmEnd = Me.tbPeriodEnd.Value
intDays = DateDiff("D", dtmBegin, dtmEnd)
strFacility_toWrite = strFacility
strSSN_toWrite = strSSN
strRUG_toWrite = DetermineRUG(blnNew, dtmAdmit, dtmAssessmentLast, strRUG, strRUGLast, "YYYY")
blnWrite = True
'End of Line
End If
ElseIf dtmAdmit < Me.tbPeriodBegin.Value Then
'Is the Assessment Date in our range of interest?
If dtmAssessment >= Me.tbPeriodBegin.Value And dtmAssessment <= Me.tbPeriodEnd.Value Then
'First(Y)Admit(Y)AdmitInRange(N)AssessInRange(Y)
dtmBegin = Me.tbPeriodBegin.Value
dtmEnd = dtmAssessment
blnClose = True
intDays = DateDiff("D", dtmBegin, dtmEnd)
strFacility_toWrite = strFacility
strSSN_toWrite = strSSN
strRUG_toWrite = DetermineRUG(blnNew, dtmAdmit, dtmAssessmentLast, strRUG, strRUGLast, "YYNY")
blnWrite = True
'End of Line
ElseIf dtmAssessment > Me.tbPeriodEnd.Value Then
'First(Y)Admit(Y)AdmitInRange(N)AssessInRange(N)
dtmBegin = Me.tbPeriodBegin.Value
dtmEnd = Me.tbPeriodEnd.Value
intDays = DateDiff("D", dtmBegin, dtmEnd)
strFacility_toWrite = strFacility
strSSN_toWrite = strSSN
strRUG_toWrite = DetermineRUG(blnNew, dtmAdmit, dtmAssessmentLast, strRUG, strRUGLast, "YYNN")
blnWrite = True
'End of Line
End If
End If
Else
'Is the Assessment Date in our range of interest?
If dtmAssessment >= Me.tbPeriodBegin.Value And dtmAssessment <= Me.tbPeriodEnd.Value Then
'First(Y)Admit(N)AdmitInRange(N)AssessInRange(Y)
dtmBegin = Me.tbPeriodBegin.Value
dtmEnd = dtmAssessment
blnClose = True
intDays = DateDiff("D", dtmBegin, dtmEnd)

```

```

strFacility_toWrite = strFacility
strSSN_toWrite = strSSN
strRUG_toWrite = DetermineRUG(blnNew, dtmAdmit, dtmAssessmentLast, strRUG, strRUGLast, "YNNY")
blnWrite = True
'End of Line
ElseIf dtmAssessment > Me.tbPeriodEnd.Value Then
    First(Y)Admit(N)AdmitInRange(N)AssessInRange(N)
    dtmBegin = Me.tbPeriodEnd.Value
    dtmEnd = Me.tbPeriodEnd.Value
    intDays = 0
    strFacility_toWrite = strFacility
    strSSN_toWrite = strSSN
    strRUG_toWrite = DetermineRUG(blnNew, dtmAdmit, dtmAssessmentLast, strRUG, strRUGLast, "YNNN")
    blnWrite = True
    'End of Line
End If
End If
End If
If blnWrite Then
    strAssessmentID_toWrite = strAssessmentId
    Call WriteResult(strFacility_toWrite, strSSN_toWrite, dtmBegin, dtmEnd, intDays, strRUG_toWrite, strAssessmentID_toWrite)
    blnWrite = False
End If
End If
strFacilityLast = strFacility
strSSNLast = strSSN
strAssessmentID_Last = strAssessmentId
blnCaidLast = blnCaid
strRUGLast = strRUG
dtmAssessmentLast = dtmAssessment
If Not IsNull(dtmDischarge) Then
    dtmDischargeLast = dtmDischarge
Else
    dtmDischargeLast = #12:00:00 AM#
End If
End If
rs.MoveNext

```

Wend  
rs.Close

```
'Finish with the last record
If blnClose And dtmDischarge = #12:00:00 AM# Then
    dtmBegin = DateAdd("d", 1, dtmAssessmentLast)
    dtmEnd = Me.tbPeriodEnd.Value
    intDays = DateDiff("D", dtmBegin, dtmEnd) + 1
    strFacility_toWrite = strFacilityLast
    strSSN_toWrite = strSSNLast
    strRUG_toWrite = strRUGLast
    strAssessmentID_toWrite = strAssessmentID_Last
    Call WriteResult(strFacility_toWrite, strSSN_toWrite, dtmBegin, dtmEnd, intDays, strRUG_toWrite, strAssessmentID_toWrite)
End If

Set rs = Nothing
Set rs2 = Nothing

DoCmd.SetWarnings (WarningsOn)
End Sub
```